## REMARKS

In response to the Office Action dated July 13, 2007, claims 1, 11-14, 16, 17, 20-26 and 28 have been amended and claims 9, 15, 18, 27, 29, and 31 have been canceled. Therefore, claims 1-8, 10-14, 16, 17, 19-26, 28, and 30 are now in the case. In light of the amendments and arguments set forth herein, reexamination and reconsideration of the application are requested.

### Specification Amendment

In reviewing the specification, the Applicants noted a minor error on page 4, line 9. In particular, the word "cached" should be "cache". The Applicants have amended the specification to correct this error.

### Section 112, Second Paragraph Rejections

The Office Action rejected claims 11-19 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter that the Applicants regard as their invention. In particular, regarding claim 11, the Office Action stated that the language ". . . lighting each of the vertices . . ." (line 6) "is considered unclear. It is not clear whether (1) the input vertices (e.g. all the vertices) are lit or (2) the vertices remaining post culling are lit." Appropriate correction was required. In addition, the Office Action stated that for "purposes of applying prior art said language is considered to read on (2)."

In response, the Applicants have amended claim 11 to further clarify that the culled vertices are not lit. In particular, claim 11now recites "lighting each of the transformed vertices using the lighting module, except for the culled transformed vertex, to compute color and generate transformed and lighted vertices from the rendering data".

Regarding claims 13-18, the Office Action stated that it was "unclear as to whether said language intends to refer to vertices (plural) or a vertex (singular) as said language seems to mix together plural and singular language (e.g., '. . . whether a vertices forms . . .'; 'discarding the vertices if it . . .', etc.)." Appropriate correction was required. Moreover,

the Office Action stated that for "purposes of applying prior art said language is considered to read on a plurality of vertexes (i.e., vertices)."

In response, the Applicants have amended claims 13, 14, 16, and 17 to further clarify that the claims are directed to a single vertex. Moreover, claims 15 and 18 have been canceled.

Therefore, based on the amendment to claims 11-14, 16, and 17, and the cancellation of claims 15 and 18, the Applicants respectfully submit that amended claims 11-14, 16, 17, and 19, are patentable under 35 U.S.C. § 112, second paragraph. The Applicants, therefore, respectfully request reexamination, reconsideration and withdrawal of the rejection of claims 11-19 under 35 U.S.C. § 112, second paragraph.

Section 101 Rejections

The Office Action rejected claims 20-27 under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter. The Office Action stated that "[W]hen nonfunctional descriptive material is recorded on some computer-readable medium, in a computer or on an electromagnetic carrier signal, it is not statutory since no requisite functionality is present to satisfy the practical application requirement. Merely claiming nonfunctional descriptive material, i.e., abstract ideas, stored in a computer-readable medium, in a computer, or on an electro-magnetic carrier signal does not make it statutory."

In response, the Applicants have amended independent claim 20 to now recite "a computer-implemented process contained on a computer-readable medium for rendering graphics on an embedded device". Accordingly, the Applicants respectfully submit that amended independent claim 20 is patentable under 35 U.S.C. § 101 based on the amendment to claim 20 and the legal and technical arguments set forth above and below.

Moreover, claims 21-27 depend from amended independent claim 20 and thus also contain patentable subject matter (MPEP § 2143.03). The Applicants, therefore,

respectfully request reexamination, reconsideration and withdrawal of the rejection of claims 20-27 under 35 U.S.C. § 101.

Section 102(e) Rejections

The Office Action rejected claims 1-5, 9-12, and 19 under 35 U.S.C. § 102(e) as being anticipated by Baker et al. (U.S. Patent No.7,190,362). The Office Action stated that Baker et al. disclose each and every element or feature recited by the Applicants' claims. In response, the Applicants respectfully traverse these rejections based on the amendments to claims 1, 11, 20, and 28, and the following legal and technical analysis.

In particular, the Applicants submit that Baker et al. is missing several features recited by the Applicants' claims. In particular, Baker et al. do not disclose, either explicitly or implicitly, the material claimed features of:

1. (Regarding amended independent claim 1): "defining a vertex cache as a software cache located within a transform and lighting module, and determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module";

2. (Regarding amended independent claim 11): "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed".

Amended Independent Claim 1

Amended independent claim 1 of the Applicants claimed invention includes a computer-implemented method for processing rendering data containing vertices. The method includes defining a vertex cache as a software cache located within a transform and lighting module, and determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting

module. The method also includes transforming vertices of the rendering data that have not already been transformed from model space into clip space, and continuing to store vertices of the rendering data that have already been transformed but not lighted in the vertex cache as needed to facilitate a single streamline branched architecture that avoids processing duplication of the vertices.

Claim 1 recites "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module". As stated in the Applicants' specification, the "vertex cache stores vertices as needed to avoid duplication in processing of the vertices" (specification, page 7, lines 8-9). The "vertex cache is implemented in software and not hardware. This alleviates the need for additional hardware that is at a premium in an embedded device" (specification, page 3, lines 24-26). The location of the vertex caches is also important. In particular, "the T&L (transform and lighting) module and pipeline includes a software-implemented vertex cache that is ***located in the T&L module***, while existing T&L techniques may include a vertex cache between the T&L layer and the rasterizer" (specification, page 7, lines 14-17; emphasis added).

As shown in FIG. 4 of the Applicants' specification, one of the ways in which the vertex cache 415 avoids duplication in processing of the vertices is by storing a vertex from the vertex buffer 410 that has already been transformed. Specifically, "a determination is made as to whether the vertex information of the input data 400 has been previously transformed. If the vertex information has been previously transformed, vertex information is send to a vertex cache 415" (specification, page 17, lines 1-3). In other words, if a vertex has been transformed but not lighted, that vertex bypasses the transformation module 420 and goes directly to the vertex cache 415 for storage.

In contrast, Baker et al. merely disclose a vertex database 411 (located **outside** of the T&L module 420) that stores only vertices that have *already been* **transformed and**

**lighted**. Specifically, as seen from FIG. 4 of Baker et al., the vertex database 411 includes a vertex array 210 and an output array 412. The processor 420 includes the T&L module. In particular, Baker et al. note that "[U]pon reading 462 the entry from vertex array 210, processor 420 **transforms and lights** the vertex in accordance with the motions of the object as defined by the software application or user" (col. 4, lines 9-12). By looking at FIG. 4 of Baker et al., it can be seen that the vertex database 411 (including the vertex array 210 and the output array 412) are **located outside** of the processor 420 (i.e., the T&L module).

In addition, the vertex database 411 of Baker et al. merely stored vertices that have already been transformed and lighted. "Once the vertex has been **transformed and lighted**, processor 420 stored the transformed vertex in output array 412 . . ." (col. 4, lines 21-24). It is impossible for the output array 412 to store a vertex that has been transformed but not lighted, because the output array 412 (and vertex database 410) are located outside of the processor 420.

The Applicants, therefore, respectfully traverse this rejection of amended independent claim 1 because Baker et al. do not teach, either explicitly or implicitly, the material claimed feature recited in claim 1 of "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module".

Because the Applicants' claim 1 recites at least one feature that is neither explicitly disclosed nor suggested by Baker et al., the Applicants respectfully submit that the rejection of amended independent claim 1 under 35 U.S.C. § 102(e) as being anticipated by Baker et al. has been overcome. Moreover, rejected claims 2-5, 9, and 10 depend from amended independent claim 1 and are therefore also novel over Baker et al. (MPEP § 2143.03). The Applicants, therefore, respectfully request reexamination, reconsideration and withdrawal of the rejection of claims 1-5, 9, and 10 under 35 U.S.C.

§ 102(e) as being anticipated by Baker et al. based on the amendments to independent claim 1 and the arguments above and below.

Amended Independent Claim 11

Amended independent claim 11 of the Applicants claimed invention includes a process for transforming and lighting rendering data. The process includes inputting rendering data in model space containing vertices, and transforming vertices in the rendering data from model space to clip space to generate transformed vertices. The process further includes culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed, and lighting each of the transformed vertices using the lighting module, except for the culled transformed vertex, to compute color and generate transformed and lighted vertices from the rendering data.

Claim 11 recites "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed". As stated in the Applicants' specification, the "culling module is positioned before the lighting module to examine each vertex and determine whether to cull or keep the vertex" (specification, page 3, lines 29-30). By ""discarding vertices that do not need to be lit or are not needed, the culling module decreases the number of vertices being processed by the lighting module. Since the lighting module is the most computationally intensive module of the transform and lighting module, this improves processing efficiency" (specification, page 4, lines 1-4; emphasis added). This means that the process recited by Applicants' claim 11 "reduces the lighting processing by culling vertices before lighting, while existing T&L techniques *transform and light* **all** vertices" (specification, page 7, lines 12-14; emphasis added).

As shown in FIG. 4 of the Applicants' specification, the culling module 425 is located after the transformation module 420 and **before** the lighting module 430. This means that the culling module 425 "is used to perform culling of the transformed vertex information prior to the information being processed by a lighting module 430"

(specification, page 17, lines 22-23). "This placement of the culling module 425 **before** the lighting module 430 reduces the number of vertex needing to be lit by the lighting module and saves both time and processing power" (specification, page 17, lines 27-29; emphasis added). In other words, the transformed vertex is culled **before** processing by the lighting module to save time and processing power.

In contrast, Baker et al. merely disclose **transforming and lighting** each vertex. No culling of any kind is performed. Specifically, "[O]nce the vertex has been **transformed and lighted**, processor 420 stored the transformed vertex in output array 412 . . ." (col. 4, lines 21-24). The Office Action stated that Baker et al. disclose "whether to cull (e.g. eliminate the storage of redundant vertex information) each [of] the vertices prior to lighting the rendering data". However, Baker et al. merely teach that the "processor 420 need only **transform and light each vertex once**, while processor 320 (the prior art processor) may be required to repeatedly transform and light the same vertices" (col. 5, lines 21-23). In other words, while Baker et al. does reduce the redundant processing of vertices, each vertex is transformed and lighted **at least once**. In contrast, Applicants' claim 11 recites "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed" and lighting each of the transformed vertices using the lighting module, except for the culled transformed vertex. Thus, some vertices (the culled vertices) are **not lit** by the lighting module.

The Applicants, therefore, respectfully traverse this rejection of amended independent claim 11 because Baker et al. do not teach, either explicitly or implicitly, the material claimed feature recited in claim 11 of "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed".

Because the Applicants' claim 11 recites at least one feature that is neither explicitly disclosed nor suggested by Baker et al., the Applicants respectfully submit that the rejection of amended independent claim 11 under 35 U.S.C. § 102(e) as being

anticipated by Baker et al. has been overcome. Moreover, rejected claims 12 and 19 depend from amended independent claim 11 and are therefore also novel over Baker et al. (MPEP § 2143.03). The Applicants, therefore, respectfully request reexamination, reconsideration and withdrawal of the rejection of claims 11, 12, and 19 under 35 U.S.C. § 102(e) as being anticipated by Baker et al. based on the amendments to independent claim 11 and the arguments above and below.

Section 103(a) Rejections

The Office Action rejected claims 6, 7, 13-18, and 20-31 under 35 U.S.C. § 103(a) as being unpatentable over Baker et al. as applied to claims 1-5, 9-12, and 19, in view of a book by Foley et al. entitled "Computer Graphics: Principles and Practice". The Office Action contended that the combination of Baker et al. and Foley et al. teach all the elements of the Applicants' claimed invention.

In response, the Applicants respectfully traverse these rejections. In general, the Applicants submit that the combination of Baker et al. and Foley et al. is lacking at least one element of the Applicants' claimed invention. More specifically, neither Baker et al. nor Foley et al. disclose, either explicitly or implicitly, the material claimed features of:

1. (Regarding amended independent claim 1): "defining a vertex cache as a software cache located within a transform and lighting module, and determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module";

2. (Regarding amended independent claim 11): "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed".

3. (Regarding amended independent claim 20): "transforming the 3D data into clip space in a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices, and performing view frustum clipping of the NHCS fixed-point format vertices to generate an output of 2D screen coordinates to render the graphics represented by the rendering data on the embedded device";

4. (Regarding amended independent claim 28): "a vertex cache implemented as a software cache and located within the transform and lighting module that stores a first vertex contained in the rendering data such that the first vertex has previously been transformed but has not previously been lighted such that the first vertex is not processed by the transformation module, a culling module positioned after the transformation module and before the lighting module that culled a second vertex from the transformed vertices prior to processing by the lighting module after determining that the second vertex was not needed such that the second vertex is not processed by the lighting module, a texture generation and texture transformation module that computes texture coordinates and transforms the texture coordinates into a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices in clip space, and a view frustum module positioned after the lighting module and after the texture generation and transformation module that performs view frustum clipping of the NHCS fixed-point format vertices in clip space to generate output data that can be rendered for display on a display device of the embedded computing device".

Further, the combination fails to appreciate the advantages of these claimed features. In addition, there is no technical suggestion or motivation disclosed in either Baker et al. or Foley et al. to define these claimed features. Thus, the Applicants submit that the combination of Baker et al. and Foley et al. cannot make obvious the Applicants'

claimed features listed above.

To make a prima facie showing of obviousness, all of the claimed features of an Applicant's invention must be considered, especially when they are missing from the prior art. If a claimed feature is not disclosed in the prior art and has advantages not appreciated by the prior art, then no prima facie showing of obviousness has been made. The Federal Circuit Court has held that it was an error not to distinguish claims over a combination of prior art references where a material limitation in the claimed system and its purpose was not taught therein. *In re Fine,* 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988). Moreover, as stated in the MPEP, if a prior art reference does not disclose, suggest or provide any motivation for at least one claimed feature of an Applicant's invention, then a prima facie case of obviousness has not been established (MPEP § 2142).

Amended Independent Claim 1

As mentioned above, amended independent claim 1 recites "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module". As also stated above, Baker et al. merely disclose a vertex database 411 located **outside** of the T&L module 420 that stores only vertices that have *already been* **transformed and lighted**.

Foley et al. adds nothing to the cited combination that would render Applicants' claim 1 obvious. Foley et al. nowhere discloses a vertex cache, must less the vertex cache recited in Applicants' claim 1.

The combination of Baker et al. and Foley et al. also fails to appreciate or recognize the advantages of the Applicants' claimed feature recited in claim 1 of "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been

transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module". More specifically, this claimed feature "alleviates the need for additional hardware that is at a premium in an embedded device" (specification, page 3, lines 24-26). Neither Baker et al. nor Foley et al. discuss or appreciate these advantages of this feature recited in Applicants' claim 1.

The Applicant, therefore, submits that obviousness cannot be established since the combination of Baker et al. and Foley et al. fails to teach, disclose, suggest or provide any motivation for the Applicants' claimed feature recited in claim 1 of "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module". In addition to explicitly lacking this feature, Baker et al. and Foley et al. fail to implicitly disclose, suggest, or provide motivation for this feature. Further, the combination fails to appreciate advantages of this claimed feature recited in claim 1.

Amended Independent Claim 11

As mentioned above, amended independent claim 11 recites "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed". As also stated above, Baker et al. merely disclose **transforming and lighting** each vertex. No culling of any kind is performed.

Foley et al. add nothing to the cited combination that would render Applicants' claim 1 obvious. Foley et al. nowhere disclose any type of culling of vertices. In addition, Foley et al. nowhere disclose culling a vertex prior to lighting after determining that the vertex is not needed.

The combination of Baker et al. and Foley et al. also fails to appreciate or recognize the advantages of the Applicants' claimed feature recited in claim 11 of "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed". More specifically, this claimed feature "decreases the number of vertices being processed by the lighting module. Since the lighting module is the most computationally intensive module of the transform and lighting module, this improves processing efficiency" (specification, page 4, lines 1-4; emphasis added). Moreover, "[T]his placement of the culling module 425 before the lighting module 430 reduces the number of vertex needing to be lit by the lighting module and saves both time and processing power" (specification, page 17, lines 27-29; emphasis added). Neither Baker et al. nor Foley et al. discuss or appreciate these advantages of this feature recited in Applicants' claim 11.

The Applicant, therefore, submits that obviousness cannot be established since the combination of Baker et al. and Foley et al. fails to teach, disclose, suggest or provide any motivation for the Applicants' claimed feature recited in claim 11 of "culling a transformed vertex of the transformed vertices prior to processing by a lighting module after determining that the transformed vertex is not needed". In addition to explicitly lacking this feature, Baker et al. and Foley et al. fail to implicitly disclose, suggest, or provide motivation for this feature. Further, the combination fails to appreciate advantages of this claimed feature recited in claim 11.

Amended Independent Claim 20

Amended independent claim 11 of the Applicants claimed invention includes a computer-implemented process for rendering graphics on an embedded device. The process includes inputting 3D data containing vertices in model space, and transforming the 3D data into clip space in a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices. The process also includes examining each of the NHCS fixed-point format vertices before lighting to determine whether to cull the NHCS fixed-point format vertices, storing the NHCS fixed-point format vertices as needed in a vertex cache to provide a single streamline

branched architecture that avoids processing duplication of the NHCS fixed-point format vertices, and performing view frustum clipping of the NHCS fixed-point format vertices to generate an output of 2D screen coordinates to render the graphics represented by the rendering data on the embedded device.

Claim 20 recites "transforming the 3D data into clip space in a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices, and performing view frustum clipping of the NHCS fixed-point format vertices to generate an output of 2D screen coordinates to render the graphics represented by the rendering data on the embedded device". As stated in the Applicants' specification, "[T]ypical graphic rendering standards (such as Direct3D) designed for desktop systems use floating-point operations for the transform and lighting process. In embedded systems, however, the CPUs may not be powerful enough to support floating-point operations and there is typically no coprocessor or GPU for accelerating the floating-point operations" (specification, page 2, lines 14-18).

Moreover, the "task module 140 inputs the raw rendering data 120 in a floating-point format and converts the data 120 into a desired fixed-point format. In some embodiments, the task module 140 is capable of converting the data 120 in a floating-point format into either a traditional fixed-point format or a preferred NHCS fixed-point format" (specification, page 7, lines 30-31 and page 8, lines 1-3). The data is in a NHCS fixed-point format and the mathematical operation and graphics functions (such as view frustum clipping) "are specially created to process the (NHCS) fixed-point data" (specification, page 8, lines 7-9).

As noted in the Office Action, Baker et al. do not disclose performing view frustum clipping or vertices in a fixed-point format in a normalized homogenous coordinate system (NHCS). The Office Action stated that Foley et al. "teaches the use of a normalized homogeneous coordinate system (p. 204-208, § 5.2; 213-217, § 5.6)". However, Foley et al. do not disclose transforming the 3D data into a fixed-point format in a normalized homogenous coordinate system (NHCS). Moreover, Foley et al. do not disclose

performing view frustum clipping of the NHCS fixed-point format vertices.

The combination of Baker et al. and Foley et al. also fails to appreciate or recognize the advantages of the Applicants' claimed feature recited in claim 20 of "transforming the 3D data into clip space in a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices, and performing view frustum clipping of the NHCS fixed-point format vertices to generate an output of 2D screen coordinates to render the graphics represented by the rendering data on the embedded device". More specifically, "processing of the data in the NHCS fixed-point format allows more efficient use of valuable memory and processing power" (specification, page 20, lines 13-15). Neither Baker et al. nor Foley et al. discuss or appreciate these advantages of this feature recited in Applicants' claim 20.

The Applicant, therefore, submits that obviousness cannot be established since the combination of Baker et al. and Foley et al. fails to teach, disclose, suggest or provide any motivation for the Applicants' claimed feature recited in claim 20 of "transforming the 3D data into clip space in a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices, and performing view frustum clipping of the NHCS fixed-point format vertices to generate an output of 2D screen coordinates to render the graphics represented by the rendering data on the embedded device". In addition to explicitly lacking this feature, Baker et al. and Foley et al. fail to implicitly disclose, suggest, or provide motivation for this feature. Further, the combination fails to appreciate advantages of this claimed feature recited in claim 20.

Amended Independent Claim 28

Amended independent claim 28 of the Applicants claimed invention includes a transform and lighting module for preparing rendering data for rendering on an embedded computing device. The module includes a transformation module that transforms vertices in the rendering data into clip space to generate transformed vertices, a vertex cache implemented as a software cache and located within the

transform and lighting module that stores a first vertex contained in the rendering data such that the first vertex has previously been transformed but has not previously been lighted such that the first vertex is not processed by the transformation module, and a lighting module that computes color for each of the transformed vertices. The module also includes a culling module positioned after the transformation module and before the lighting module that culled a second vertex from the transformed vertices prior to processing by the lighting module after determining that the second vertex was not needed such that the second vertex is not processed by the lighting module, a texture generation and texture transformation module that computes texture coordinates and transforms the texture coordinates into a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices in clip space, and a view frustum module positioned after the lighting module and after the texture generation and transformation module that performs view frustum clipping of the NHCS fixed-point format vertices in clip space to generate output data that can be rendered for display on a display device of the embedded computing device.

Claim 28 recites "a vertex cache implemented as a software cache and located within the transform and lighting module that stores a first vertex contained in the rendering data such that the first vertex has previously been transformed but has not previously been lighted such that the first vertex is not processed by the transformation module, a culling module positioned after the transformation module and before the lighting module that culled a second vertex from the transformed vertices prior to processing by the lighting module after determining that the second vertex was not needed such that the second vertex is not processed by the lighting module, a texture generation and texture transformation module that computes texture coordinates and transforms the texture coordinates into a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices in clip space, and a view frustum module positioned after the lighting module and after the texture generation and transformation module that performs view frustum clipping of the NHCS fixed-point format vertices in clip space to generate output data that can be rendered for display on a display device of the embedded computing device".

As stated above with regard to claim 1, Baker et al. merely disclose a vertex database 411 located **outside** of the T&L module 420 that stores only vertices that have *already been* **transformed and lighted**. Moreover, Foley et al. nowhere discloses a vertex cache.

As stated above with regard to claim 11, Baker et al. merely disclose **transforming and lighting** each vertex. No culling of any kind is performed. Moreover, Foley et al. nowhere disclose any type of culling of vertices.

As stated above with regard to claim 20, Baker et al. do not disclose performing view frustum clipping or vertices in a fixed-point format in a normalized homogenous coordinate system (NHCS). Moreover, Foley et al. do not disclose transforming the 3D data into a fixed-point format in a normalized homogenous coordinate system and performing view frustum clipping of the NHCS fixed-point format vertices.

The combination of Baker et al. and Foley et al. also fails to appreciate or recognize the advantages of the Applicants' claimed features recited in claim 28. As discussed above, these features alleviate "the need for additional hardware that is at a premium in an embedded device" (specification, page 3, lines 24-26), "reduces the number of vertex needing to be lit by the lighting module and saves both time and processing power" (specification, page 17, lines 27-29), and "processing of the data in the NHCS fixed-point format allows more efficient use of valuable memory and processing power" (specification, page 20, lines 13-15). Neither Baker et al. nor Foley et al. discuss or appreciate these advantages of this feature recited in Applicants' claim 28.

The Applicants, therefore, submits that obviousness cannot be established since the combination of Baker et al. and Foley et al. fails to teach, disclose, suggest or provide any motivation for the Applicants' claimed features recited in claim 28 of "a vertex cache implemented as a software cache and located within the transform and lighting module

that stores a first vertex contained in the rendering data such that the first vertex has previously been transformed but has not previously been lighted such that the first vertex is not processed by the transformation module, a culling module positioned after the transformation module and before the lighting module that culled a second vertex from the transformed vertices prior to processing by the lighting module after determining that the second vertex was not needed such that the second vertex is not processed by the lighting module, a texture generation and texture transformation module that computes texture coordinates and transforms the texture coordinates into a fixed-point format in a normalized homogenous coordinate system (NHCS) to obtain NHCS fixed-point format vertices in clip space, and a view frustum module positioned after the lighting module and after the texture generation and transformation module that performs view frustum clipping of the NHCS fixed-point format vertices in clip space to generate output data that can be rendered for display on a display device of the embedded computing device". In addition to explicitly lacking these features, Baker et al. and Foley et al. fail to implicitly disclose, suggest, or provide motivation for these features. Further, the combination fails to appreciate advantages of these claimed features recited in claim 28.

Therefore, as set forth in *In re Fine* and MPEP § 2142, the combination of Baker et al. and Foley et al. cannot render the Applicants' claimed invention recited in claims 1, 11, 20, and 28, obvious because Baker et al. and Foley et al. are missing at least one material feature recited in Applicants' claims 1, 11, 20, and 28, as discussed above. Consequently, because a prima facie case of obviousness cannot be established due to the lack of "some teaching, suggestion, or incentive supporting the combination", the rejection must be withdrawn. ACS Hospital Systems, Inc. v. Montefiore Hospital, 732 F.2d 1572, 1577, 221 USPQ 929, 933 (Fed. Cir. 1984); MPEP 2143.01.

Accordingly, the Applicants respectfully submit that amended independent claims 1, 11, 20, and 28 are patentable under 35 U.S.C. § 103(a) over Baker et al. in view of Foley et al. based on the amendments to claims 1, 11, 20, and 28, and the legal and technical arguments set forth above and below. Moreover, claims 6 and 7 depend from amended independent claim 1, claims 13-18 depend from amended independent claim 11, claims

21-27 depend from amended independent claim 20, and claims 29-31 depend from amended independent claim 28, and are also nonobvious over Baker et al. in view of Foley et al. (MPEP § 2143.03). The Applicants, therefore, respectfully requests reexamination, reconsideration and withdrawal of the rejection of claims 6, 7, 13-18, and 20-31 under 35 U.S.C. § 103(a) as being unpatentable over Baker et al. as applied to claims 1-5, 9-12, and 19, in view of Foley et al.

---

The Office Action rejected claim 8 under 35 U.S.C. § 103(a) as being unpatentable over Baker et al. as applied to claims 1-5, 9-12, and 19, in view of Wang et al. (U.S. Patent No. 7,139,005). The Office Action contended that the combination of Baker et al. and Wang et al. teach all the elements of the Applicants' claimed invention.

In response, the Applicants respectfully traverse this rejection. In general, the Applicants submit that the combination of Baker et al. and Wang et al. is lacking at least one element of the Applicants' claimed invention. More specifically, neither Baker et al. nor Wang et al. disclose, either explicitly or implicitly, the material claimed feature of "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module".

Further, the combination fails to appreciate the advantages of these claimed features. In addition, there is no technical suggestion or motivation disclosed in either Baker et al. or Wang et al. to define these claimed features. Thus, the Applicants submit that the combination of Baker et al. and Wang et al. cannot make obvious the Applicants' claimed feature recited in claim 1.

As also stated above, Baker et al. merely disclose a vertex database 411 located **outside** of the T&L module 420 that stores only vertices that have ***already been transformed and lighted***. Wang et al. add nothing to the cited combination that would

render Applicants' claim 1 obvious. Wang y et al. nowhere discloses a vertex cache as recited in Applicants' claim 1.

The combination of Baker et al. and Wang et al. also fails to appreciate or recognize the advantages of the Applicants' claimed feature recited in claim 1 of "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module". More specifically, this claimed feature "alleviates the need for additional hardware that is at a premium in an embedded device" (specification, page 3, lines 24-26). Neither Baker et al. nor Wang et al. discuss or appreciate these advantages of this feature recited in Applicants' claim 1.

The Applicant, therefore, submits that obviousness cannot be established since the combination of Baker et al. and Wang et al. fails to teach, disclose, suggest or provide any motivation for the Applicants' claimed feature recited in claim 1 of "defining a vertex cache as a software cache located within a transform and lighting module" and "determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a transformation module of the transform and lighting module". In addition to explicitly lacking this feature, Baker et al. and Wang et al. fail to implicitly disclose, suggest, or provide motivation for this feature. Further, the combination fails to appreciate advantages of this claimed feature recited in claim 1.

The Applicants, therefore, submits that obviousness cannot be established since the combination of Baker et al. and Wang et al. fails to teach, disclose, suggest or provide any motivation for the Applicants' claimed features recited in claim 1 of "defining a vertex cache as a software cache located within a transform and lighting module, and determining that a first vertex of the rendering data has already been transformed but not lighted and storing the first vertex in the vertex cache such that the first vertex bypasses a

transformation module of the transform and lighting module". In addition to explicitly lacking this feature, Baker et al. and Wang et al. fail to implicitly disclose, suggest, or provide motivation for this feature. Further, the combination fails to appreciate advantages of this claimed feature recited in claim 1.

Therefore, as set forth in *In re Fine* and MPEP § 2142, the combination of Baker et al. and Wang et al. cannot render the Applicants' claimed invention recited in claim 1 obvious because Baker et al. and Wang et al. are missing at least one material feature recited in Applicants' claim 1, as discussed above. Consequently, because a prima facie case of obviousness cannot be established due to the lack of "some teaching, suggestion, or incentive supporting the combination", the rejection must be withdrawn. ACS Hospital Systems, Inc. v. Montefiore Hospital, 732 F.2d 1572, 1577, 221 USPQ 929, 933 (Fed. Cir. 1984); MPEP 2143.01.

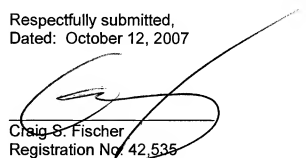
Accordingly, the Applicants respectfully submit that amended independent claim 1 is patentable under 35 U.S.C. § 103(a) over Baker et al. in view of Wang et al. based on the amendments to claim 1 and the legal and technical arguments set forth above and below. Moreover, claim 8 depends from amended independent claim 1 and is also nonobvious over Baker et al. in view of Wang et al. (MPEP § 2143.03). The Applicants, therefore, respectfully requests reexamination, reconsideration and withdrawal of the rejection of claim 8 under 35 U.S.C. § 103(a) as being unpatentable over Baker et al. as applied to claims 1-5, 9-12, and 19, in view of Wang et al.

## Conclusion

In view of the amendments to claims 1, 11-14, 16, 17, 20-26 and 28, the cancellation of claims 9, 15, 18, 27, 29, and 31, and the arguments set forth above, the Applicants submit that pending claims 1-8, 10-14, 16, 17, 19-26, 28, and 30 are in condition for immediate allowance. The Examiner, therefore, is respectfully requested to withdraw the outstanding rejections of the claims and to pass each of the pending claims of this application to issue.

In an effort to expedite and further the prosecution of the subject application, the Applicants kindly invite the Examiner to telephone the Applicants' attorney at (805) 278-8855 if the Examiner has any comments, questions or concerns, wishes to discuss any aspect of the prosecution of this application, or desires any degree of clarification of this response.

Respectfully submitted,
Dated: October 12, 2007



Craig S. Fischer
Registration No. 42,535
Attorney for Applicants

LYON & HARR, L.L.P.
300 East Esplanade Drive, Suite 800
Oxnard, CA 93036-1274
Tel: (805) 278-8855
Fax: (805) 278-8064